

Web Service LPR ISS

Version 3.5 (internal use)

Complete service description:

Route `"/set-view?recognizers=high"`

- Only uses the parameter "recognizers" and only one possible value "high"(high speed ones)
- Returns all the configured plate recognizers LPR
- Response example:

```
{
  "notification": "recognizers",
  "structure": [
    {
      "name": "Plate Recognizer 2",
      "id": "2",
      "latitude": "19.425371",
      "longitude": "-99.196082",
      "idCamera": 2
    },
    {
      "name": "Plate Recognizer 1",
      "id": "1",
      "latitude": "19.394629",
      "longitude": "-99.173401",
      "idCamera": 1
    }
  ],
  "reception": "ok",
  "details": "exception",
  "guid": "3a254bc7-5d5e-44eb-ae73-cdd4917a793b"
}
```

*****NOTE:** The recognizers service is now using a coordinates database "coordinates" which has the following structure:

Database name: "coordinates"

Table name: "coordinates"

Table creation query:

```
CREATE TABLE coordinates
(
  id integer,
  latitude text,
  longitude text
)
WITH (
  OIDS=FALSE
);
ALTER TABLE coordinates
OWNER TO postgres;
```

Route

`/new-plate?plate=XXXXXX&incidentFolio=123456789&database=blacklist`

`/new-plate?plate=XXXXXX&incidentFolio=123456789&database=whitelist`

- Uses a plate parameter “plate” which must have any alphanumeric character whether if its upper case or not, the “database” parameter, and an additional field for comment “incidentFolio”
- If database parameter is “blacklist”, it will be added to the blacklist database
- If database parameter is “whitelist”, it will be added to the blacklist database
- Response example:

```
{
  "reception": "ok",
  "guid": "3e8e7854-b7db-4218-af66-0e3cb3f8dfa8"
}
{
  "reception": "nok",
  "details": "exception",
  "guid": "3e8e7854-b7db-4218-af66-0e3cb3f8dfa8"
}
```

Route `/erase-plate?plate=XXXXXX&database=whitelist`

`/erase-plate?plate=XXXXXX&database=blacklist`

- Uses a plate parameter “plate” which must have any alphanumeric character whether if its upper case or not, and the “database” parameter
- If database parameter is “blacklist”, it will be deleted from the blacklist database
- If database parameter is “blacklist”, it will be deleted from the blacklist database

- Response example:

```
{
  "reception": "ok",
  "guid": "3e8e7854-b7db-4218-af66-0e3cb3f8dfa8"
}

{
  "reception": "nok",
  "details": "exception",
  "guid": "3e8e7854-b7db-4218-af66-0e3cb3f8dfa8"
}
```

Route `"/get-plate?plate=XXXXXX&&initialDate=YYYYMMDDTHHMISS&endDate=YYYY-MM-DD HH-MI-SS&idRecognizer=X&justDate=boolean"`

- Uses the parameters:
 - Search into the auto database
 - "plate" (plate) which must have any alphanumeric character whether if its upper case or not
 - "initialDate" (initial timestamp) using the standard timestamp format **ISO8601***
 - "endDate" (final timestamp) using the standard timestamp format **ISO8601***
 - "idRecognizer" which has an integer value to filter that certain recognizer id.
 - "justDate" Boolean (true, false). If "true" the query response won't include the image, otherwise it will send base64 image for each matching result.

NOTES: THE UNIQUE VALUE NEEDED FOR THIS SERVICE IS THE "plate" (PLATE) ONE, GIVEN THAT CASE, THE SERVICE WILL RESPONSE AT MOST 100 RESULTS FROM ALL THE RECOGNIZERS, SORTED FROM THE OLDEST TO THE NEWEST IN A 30 DAYS INTERVAL FROM THE TIME IT IS QUERIED. (100 results was a good bandwidth practice, however this unfiltered query could be changed as convenient).

FOR THIS SERVICE, THE PLATE PARAMETER CAN BE SEND INCOMPLETE, MEANING IT WILL MATCH THE PLATES THAT MATCH AT LEAST WHAT IS WRITTEN AS A PARAMETER.

- Response examples:

```
{
  "structure": [
    {
      "tid": "59355",
      "name": "License plate recognizer 1",
      "timeStamp": "15/11/2017 10:10:46",
      "plate": "MNF9871",
      "direction": "leaving",
      "comment": "Comment",
      "image": "BASE64IMAGE"
    },
    {
      "tid": "59357",
      "name": "License plate recognizer 1",
      "timeStamp": "15/11/2017 10:11:07",
      "plate": "MEU6378",
      "direction": "leaving",
      "comment": "Comment",
      "image": " BASE64IMAGE "
    }
  ],
  "reception": "ok",
  "guid": "290fe772-6d64-4ad8-8615-14e3126621a2"
}
```

-

```
{
  "structure": [
    {
      "tid": "59355",
      "name": "License plate recognizer 1",
      "timeStamp": "15/11/2017 10:10:46",
      "plate": "MNF9871",
      "direction": "leaving",
```

```

    "comment": "Comment"
  },
  {
    "tid": "59357",
    "name": "License plate recognizer 1",
    "timeStamp": "15/11/2017 10:11:07",
    "plate": "MEU6378",
    "direction": "leaving",
    "comment": "Comment"
  }
],
"reception": "ok",
"guid": "290fe772-6d64-4ad8-8615-14e3126621a2"
}

```

Route `"/get-whitelist"`

- Does not require any parameter
- Returns all the plates stored in whitelist database
- Response example:

```

{
  "plates": [
    "MRT1368"
  ],
  "reception": "ok",
  "guid": "87c1a3f9-a0a1-4fe6-b476-860bda1c4d4a"
}

```

Route `"/get-blacklist"`

- Does not require any parameter
- Returns all the plates stored in blacklist database
- Response example:

```

{
  "plates": [
    "MRT1368",
    "ABC321"
  ],
  "reception": "ok",
  "guid": "87c1a3f9-a0a1-4fe6-b476-860bda1c4d4a"
}

```

Route `"/insert-comments?comment=Plate Test&id=34597"`

- Inserts a comment into the LPR register specified
- Uses the following parameters:
- `"comment"`, where the comment is to be written
- `"id"`: which is to be the database register id for the specific recognition stored, for instance this `"id"` is given when using the `"/get-plate"` service
- Response example:

```
{  
  "reception": "ok",  
  "guid": "3e928866-02d8-4cdf-acea-34c2dae4a116"  
}
```

For all these services, if given insufficient parameters, this immediate response will be returned:

```
{  
  "notification": "Error",  
  "details": "Input data missing"  
}
```